



COURS DE DELPHI

SÉRIE N° 02

OBJECTIF PÉDAGOGIQUE :

À la fin de cette série, le stagiaire doit être capable de créer et d'exécuter un projet Delphi.

PLAN DE LA LEÇON :

INTRODUCTION

I- OPÉRATIONS SUR UNE NOUVELLE APPLICATION DELPHI

- 1- Création d'un projet
- 2- Enregistrer le projet
- 3- Exécution de l'application

QUESTIONS DE COURS

INTRODUCTION :

Après son lancement, Delphi se présente sous la forme de 4 fenêtres :

- A-** La première fenêtre occupe la partie supérieure de l'écran. Elle correspond à l'environnement de programmation proprement dit. Cette fenêtre contient :
 - La barre de titre ;
 - La barre de menu de Delphi ;
 - Une zone « barre d'outils » (sur la gauche) ;
 - Une zone contenant les divers composants regroupés par familles.
- B-** La seconde fenêtre se trouve par défaut à gauche de l'écran : c'est l'inspecteur d'objets. Il permet de visualiser, pour chaque objet ou composant, les propriétés et les événements auxquels l'objet peut répondre.
- C-** La troisième fenêtre constitue la fiche principale de la future application Delphi. Il s'agit, au départ, d'une fenêtre vide dans laquelle on placera les divers objets.
- D-** La dernière fenêtre, cachée sous la précédente constitue l'éditeur proprement dit, contenant le code source de l'application.

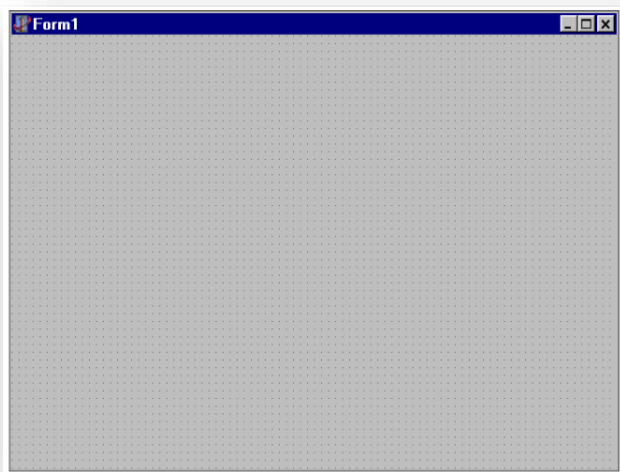
I- OPÉRATIONS SUR UNE NOUVELLE APPLICATION DELPHI

1- Création d'un projet :

Pour démarrer une nouvelle application (projet), il faut choisir l'option **New Application** du menu **File**.

Un projet est un l'ensemble des fichiers nécessaires au fonctionnement de l'application.

Quand on charge Delphi, un projet de base est créé. Il comporte une fenêtre prête à recevoir différents objets.



Il est évident que cette première application n'est pas encore exécutable. Pour ce faire, il est nécessaire de compiler (traduire en binaire). Le système s'est chargé de créer le code du programme (.PAS).

Explications des lignes de code :

Unit Unit1 :

Nom du module source correspondant à la fiche "Form1", ne pas changer ce nom directement, mais au moment de l'enregistrement.

Interface entête pour désigner la section qui sera "vue" par les autres modules

Uses permet d'inclure d'autres modules

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs ; nom des modules inclus

Typedébut de la déclaration du type des objets

TForm1 = class(TForm) déclaration d'un nouveau type d'objet de type fenêtre

Private {Déclarationsprivées} propriétés et méthodes "privées" du type d'objet

Public{Déclarations publiques} propriétés et méthodes "publiques" du type d'objet; elles seront visibles par les autres modules

End; fin de déclaration d'objet

var "achat" de variables globales

Form1: TForm1; instantiation de l'objet: allocation de mémoire vive

Implementation dans cette section seront écrites les méthodes des objets manipulés

{ \$R *.DFM } directive de compilation permettant d'inclure les définitions des fenêtres (.DFM)

End. Fin du module

Ce module ne peut fonctionner seul. Un module de projet (.DPR) se charge de fédérer l'ensemble des modules, mais il n'est pas visible en ce moment. Pour le visualiser, ouvrez le menu "Projet" et choisissez "voir la source". Un onglet correspondant apparaît dans l'éditeur de code.

Program Project1; nom du programme

Uses modules utilisés Forms,

Unit1 in 'UNIT1.PAS' {Form1}; c'est le module de la fiche "Form1"

{\$R *.RES} directive de compilation permettant d'inclure les fichiers ressources (.RES)

Begin début du programme principal

Application.Initialize; initialisation du programme

Application.CreateForm(TForm1, Form1); création de la fiche "Form1" de type TForm1

Application.Run; exécution de l'application

End. Fin du programme principal

On peut également visualiser le code de la fiche (.DFM), grâce à l'option "voir comme texte" de son menu contextuel.

Object Form1: TForm1

Left = 295

Top = 132

Width = 696

Height = 480

Caption = 'Form1'

Color = clBtnFace

Font.Charset = DEFAULT_CHARSET

Font.Color = clWindowText

Font.Height = -13

Font.Name = 'MS Sans Serif'

Font.Style = []

OldCreateOrder = False

PixelsPerInch = 120

TextHeight = 16

End

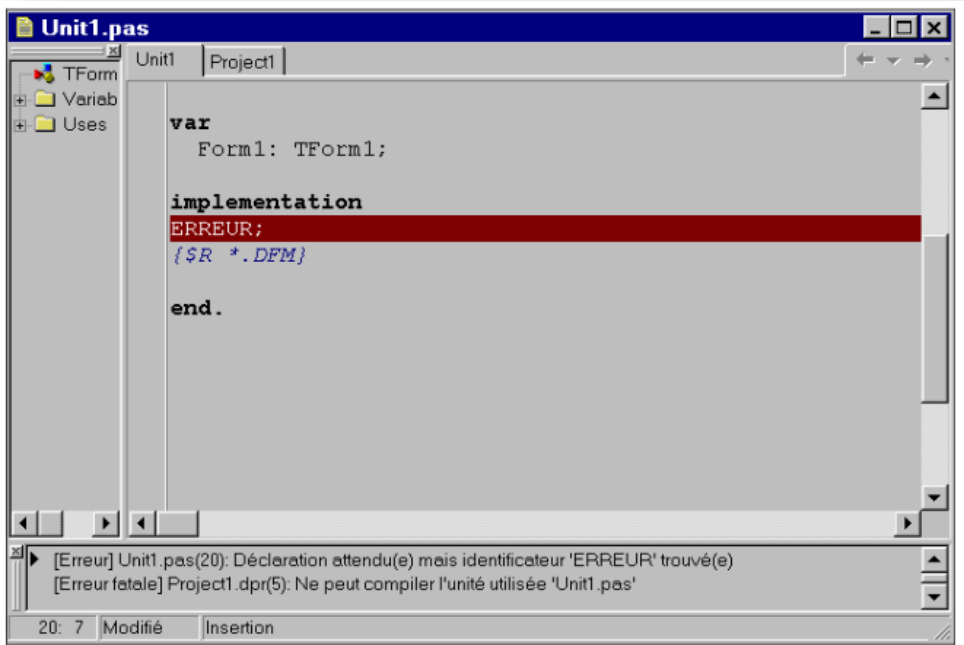
Revenir à la fiche avec "voir comme fiche" du menu contextuel.

2- Enregistrement du projet :

Pour enregistrer le projet, choisir "enregistrer projet sous" dans le menu fichier. Il faudra donner un nom aux différents modules (.PAS) et au projet (DPR).

3- Exécution de l'application :

On peut vérifier la syntaxe du code, compiler ou exécuter directement le projet. Pour disposer d'un exécutable, il est nécessaire de compiler. En cas d'erreur, la compilation s'arrête et visualise un message d'erreur.



Si l'on désire des renseignements sur l'erreur, il suffit cliquer sur le message d'erreur ou d'avertissement et d'appuyer sur la touche F1 (aide en ligne) et on obtient la fenêtre suivante :

Il suffit de rectifier le code et de relancer une compilation. Ensuite,

29. <Token1> attendu mais <token2> trouvé

Liste complète des messages d'erreur du compilateur

Il s'agit d'une erreur de syntaxe. Il y a probablement une erreur de typo dans les sources ou quelque chose a été oublié. Lorsque l'erreur se produit au début d'une ligne, l'erreur actuelle est souvent sur la ligne précédente.

```
program Produce;
var
  I: Integer
begin
    (*<-- Message d'erreur ici: ';' attendu mais 'BEGIN' trouvé*)
end.
```

Après le type Integer, le compilateur attend de trouver un point-virgule pour terminer la déclaration de variable. Il ne trouve pas de point-virgule sur la ligne courante, ainsi il lit et trouve le mot clé 'begin' au début de la ligne suivante. Il sait alors que quelque chose est faux...

```
program Solve;
var
  I: Integer;    (*point-virgule manquant*)
begin
end.
```

Dans ce cas, seul le point-virgule était manquant - cas fréquent en pratique. En général, regardez la ligne où apparaît le message d'erreur, et la ligne au-dessus pour trouver s'il manque quelque chose ou si quelque chose est mal orthographié.

On peut directement utiliser l'exécutable généré (.EXE). Quitter Delphi et lancer le programme depuis le l'explorateur Windows.

Examen des fichiers générés :

Suffixe	Signification
dof	options de compilation et informations de versions
dpr	source du projet
exe	programme exécutable compilé
dcu	unité compilée
dfm	définition de la feuille
pas	source pascal
res	ressources: icône de l'application
cfg	configuration du compilateur
d dp	diagrammes

Pour examiner certains de ces fichiers, il est pratique d'utiliser l'explorateur Windows. Mettre un raccourci vers le bloc-notes (notepad.exe) dans le répertoire SendTo.

QUESTIONS DE COURS :

Quelles sont les fichiers créés dans un projet Delphi ?

Créer une application nommée «ma première application»?

RÉPONSE :

Voir le cours